УДК 663.5:004.896

DEVELOPMENT OF MATHEMATICAL MODELS FOR VIRTUAL SENSORS TO MONITOR DYNAMIC PROCESSES IN THE DISTILLATION DEPARTMENT OF ETHANOLPRODUCTION

O. Omelchenko, N. Lutska
National University of Food Technologies

Key words:
virtual sensors,
modeling,
machine learning models,
ethanol industry,
distillation

Article history:
Received 26.11.2024
Received in revised form 28.11.2024
Accepted 16.12.2024

Corresponding author:
alexmir_98@ukr.net

ABSTRACT

The article investigates the development and comparison of mathematical models' effectiveness for virtual sensors used in mo¬nitoring processes of the distillation department in ethanol produc¬tion. The research relevance is determined by the complexity of controlling distillation parameters due to nonlinear and inertial re¬lationships between technological variables that affect the final pro¬duct quality and production energy efficiency. Effective control of these parameters is critical for optimizing production, minimizing energy consumption, and ensuring consistent product quality.

An analysis of current research regarding the application of virtual sensors in industry and machine learning methods for time series data processing has been conducted. Virtual sensors offer a flexible and adaptive solution for enhancing process control by providing real-time monitoring, predictive capabilities, and fault-tolerant operation. Based on the production data from the distillation department over 12 hours of operation, a comparative analysis of five machine learning models was performed: XGBoost, Decision Tree, NARX, Gradient Boosting, and RNN. The models' effectiveness was evaluated using MSE, R^2 metrics, and the Akaike information criterion. This approach ensures a comprehensive evaluation of both the predictive accuracy and the complexity of the models. For cross-validation, the TimeSeriesSplit method was applied, taking into account the temporal sequence of data.

According to the research results, the Gradient Boosting model demonstrates the best balance between prediction accuracy (MSE = 0.00048, $R^2 = 0.972$) and model complexity according to the Akaike criterion. Its performance stability across various evaluation datasets further validates its robustness and suitability for practical applications in an industrial setting. The model showed stable performance on all test datasets and is recommended for implementation in the distillation department's monitoring system as a component of the digital twin. By integrating this model into the digital twin framework, the distillation process can benefit from enhanced monitoring, optimization, and decision-making capabilities, ultimately improving operational efficiency and product quality.

DOI: 10.24263/2225-2916-2024-36-9

РОЗРОБКА МАТЕМАТИЧНИХ МОДЕЛЕЙ ВІРТУАЛЬНИХ СЕНСОРІВ ДЛЯ МОНІТОРИНГУ ДИНАМІЧНИХ ПРОЦЕСІВ У ДИСТИЛЯЦІЙНОМУ ВІДДІЛЕННІ СПИРТОВОГО ВИРОБНИЦТВА

О. С. Омельченко, аспірант
Н. М. Луцька, д-р техн. наук, ORCID ID: 0000-0001-8593-0431
Національний університет харчових технологій

У статті досліджується розробка та порівняння ефективності математичних моделей віртуальних сенсорів для моніторингу процесів дистиляційного відділення спиртового виробництва.
На основі виробничих даних дистиляційного відділення за 12 год роботи здійснено порівняльний аналіз п'яти моделей машинного навчання: XGBoost, DecisionTree, NARX, GradientBoosting та RNN. Для оцінки ефективності моделей використано метрики MSE, R^2 та інформаційний критерій Акаіке.
За результатами дослідження встановлено, що модель GradientBoosting демонструє найкращий баланс між точністю прогнозування (MSE=0,00048, R^2=0,972) та складністю моделі згідно з критерієм Акаіке.

Ключові слова: *віртуальні сенсори, моделювання, моделі машинного навчання, спиртова промисловість, дистиляція.*

Постановка проблеми. Дистиляція є одним із фундаментальних етапів спиртового виробництва, що визначає якість, чистоту та міцність кінцевого продукту, впливаючи на ефективність виробництва спирту в цілому. Процес дистиляції характеризується наявністю складних, нелінійних та інерційних взаємозв'язків між технологічними змінами, що ускладнює традиційним методам аналізу контроль над стабільністю і точністю контролю параметрів процесу, які визначають якість кінцевого продукту, енергетичну ефективність та загальну продуктивність системи. Застосування віртуальних сенсорів (ВС) покращує контроль за перебігом процесів шляхом моніторингу роботи фізичних сенсорів, компенсації їх функціоналу внаслідок збоїв або відмов, отримання точних значень параметрів, які важко або неможливо виміряти безпосередньо, прогнозування оптимальних режимів роботи обладнання та змін параметрів у реальному часі. Гнучкість та адаптивність ВС дає їм змогу забезпечувати стабільну роботу навіть при зміні виробничих умов, наприклад, при зміні сировини. Крім цього, вони є важливою складовою цифрових двійників, які допомагають моделювати, оптимізувати та керувати виробничими процесами у віртуальному середовищі.

Ефективність ВС значною мірою залежить від точності й адекватності математичних моделей, на основі яких вони функціонують. Ці моделі повинні враховувати складні нелінійні взаємозв'язки між параметрами процесу дистиляції, а також динамічні зміни, які виникають під час роботи обладнання. Проблема моделювання ВС для дистиляційного відділення спиртового виробництва недостатньо представлена в науковій літературі та потребує додаткового вивчення.

Аналіз останніх досліджень і публікацій. Технологічний розвиток значно розширив можливості використання віртуальних сенсорів (ВС) завдяки вдосконаленню апаратного забезпечення, алгоритмів обробки даних і доступності технологій моделювання. Це сприяло підвищенню точності, адаптивності та швидкості їх роботи, а

також активному дослідженню й впровадженню ВС у склад інформаційних систем (ІС) [1].

У працях [3—5] розглядаються ефективність та переваги ВС, у [6—11] описується їх інтеграція в різноманітні системи для покращення моніторингу та аналізу даних. Зокрема, у [2] акцентується увага на ролі ВС у подоланні розриву між фізичним і цифровим світом. ВС визначаються як програмні об'єкти, що збирають, об'єднують і агрегують дані з фізичних чи віртуальних датчиків для забезпечення інформацією, що може бути використана в ІС.

Одним із ключових аспектів функціонування ВС є застосування функції злиття даних (data fusion function), яка перетворює вихідні сигнали на потрібні змінні з використанням математичних моделей. При виборі моделей для ВС важливо забезпечити баланс між відповідністю характеристикам об'єкта керування та складністю моделі, адже простіші моделі забезпечують вищу надійність і менші витрати на їх підтримку [1].

Окрему увагу приділено використанню моделей машинного навчання у складі ВС. У [10—12] досліджується їх впровадження для підвищення можливостей ВС, зокрема для обробки часових рядів [10]. Дослідження [13] описує застосування регресійних моделей для аналізу масових витрат і температур у системах кондиціонування, що дає змогу ефективно виявляти несправності обладнання.

Всупереч значному прогресу у використанні віртуальних сенсорів у різних галузях проблеми, пов'язані із застосуванням та моделюванням ВС для дистиляційного відділення, залишаються недостатньо вивченими та потребують подальшого дослідження.

Мета дослідження: розробити та порівняти ефективність математичних моделей віртуальних аналізаторів для моніторингу процесів дистиляційного відділення спиртового виробництва в умовах динамічних змін процесу.

Матеріали і методи. ВС відкривають нові можливості для моніторингу та управління дистиляційними процесами, дають змогу підвищити точність вимірювання важливих параметрів, компенсувати збої фізичних сенсорів, прогнозувати оптимальні режими роботи та забезпечувати адаптацію до змін виробничих умов. Важливим аспектом їх успішного функціонування є вибір математичних моделей, здатних адекватно враховувати нелінійні зв'язки та динаміку системи.

Методи машинного навчання є перспективним інструментом для вирішення цих проблем завдяки можливості виявляти приховані залежності між параметрами процесу і точно прогнозувати вихідні величини на основі вхідних даних часових рядів. Використання алгоритмів, здатних ефективно працювати з нелінійними взаємозв'язками, часовими залежностями та багатовимірними даними, відкриває можливості для точного моделювання складних систем і врахування багатофакторних впливів.

Для порівняння результатів передбачення математичних моделей було використано такі показники:

1. MSE (Mean Squared Error) — метрика, яка використовується для оцінки якості прогнозу моделі, особливо в задачах регресії. MSE показує середнє значення квадратів помилок між фактичними значеннями (y_i) і прогнозованими значеннями ($\hat{y}_i$):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2,$$

де n — кількість спостережень, y_i — фактичне (реальне) значення, $\hat{y}_i$ — прогнозоване значення, $(y_i - \hat{y}_i)^2$ — квадрат помилки для кожного спостереження.

2. R^2 (коефіцієнт детермінації) — статистична метрика, яка показує, наскільки добре модель пояснює варіацію цільової змінної порівняно з її середнім значенням:

$$R^2 = 1 - \frac{RSS}{TSS},$$

де $RSS = \sum(y_i - \hat{y}_i)^2$ — залишкова сума квадратів (помилки моделі), $TSS = \sum(y_i - \bar{y})^2$ — загальна сума квадратів (варіація фактичних значень навколо середнього), y_i — фактичне значення, $\hat{y}_i$ — прогнозоване значення, $\bar{y}$ — середнє значення фактичних значень.

3. AIC (Akaike Information Criterion) — оцінювач похибки поза вибіркового передбачування і, відтак, відносної якості статистичних моделей для заданого набору даних. AIC дає змогу порівнювати моделі та обирати ту, яка найкраще збалансовує точність (якість підгонки) та простоту (число параметрів):

$$AIC = 2k - 2ln(L),$$

де k — кількість параметрів у моделі, L — максимальне значення функції правдоподібності моделі (міра того, наскільки добре модель описує дані).

Для підвищення якості перевірки роботи моделей в роботі застосовується Time Series Split — техніка перехресної перевірки (cross-validation), яка спеціально адаптована для роботи з часовими рядами. Її використання допомагає забезпечити коректне розбиття даних з урахуванням їх послідовності, що важливо для даних часових рядів, які залежать від часу. На відміну від стандартного k-foldcross-validation, де дані діляться випадковим чином, у Time Series Split дані розділяються таким чином, щоб більш ранні часові точки використовувалися для тренування, а більш пізні — для тестування. Це імітує реальний сценарій, коли прогнозується майбутнє на основі минулого.

Викладення основних результатів дослідження. На основі даних роботи дистиляційного відділення спиртового заводу (рис. 1) було проведено дослідження та порівняльний аналіз моделей для ВС. Цільовим показником обрано точність передбачення поведінки змінної за допомогою математичної моделі.

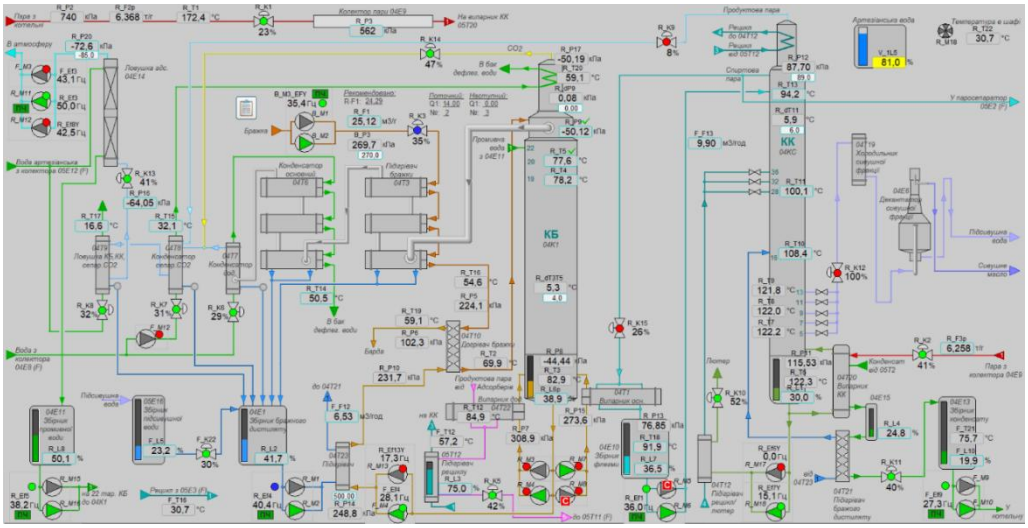

Рис. 1. Мнемосхема дистиляційного відділення

1. Дані та мета прогнозування. Розробка ВС базується на аналізі даних часових рядів, отриманих від датчиків дистиляційного відділення, які включають вимірювання температури, тиску, рівнів і витрат за 12 год роботи в стаціонарному режимі. Метою є створення моделі, здатної передбачати температуру над тринадцятою тарілкою колони концентрації (рис. 2), враховуючи високу кореляцію між змінними та наявність нестаціонарних ефектів у часових рядах. Віртуальний сенсор дає змогу доповнити фізичні датчики, а також забезпечує безперервний моніторинг та прогнозування у реальному часі, що підвищує ефективність управління технологічними процесами та якість продукції. Доповнення фізичних датчиків віртуальними дає змогу контролювати технологічні змінні у разі зникнення лінії зв'язку або поломки датчика, підвищуючи надійність системи за рахунок резервування та перехресної перевірки даних.

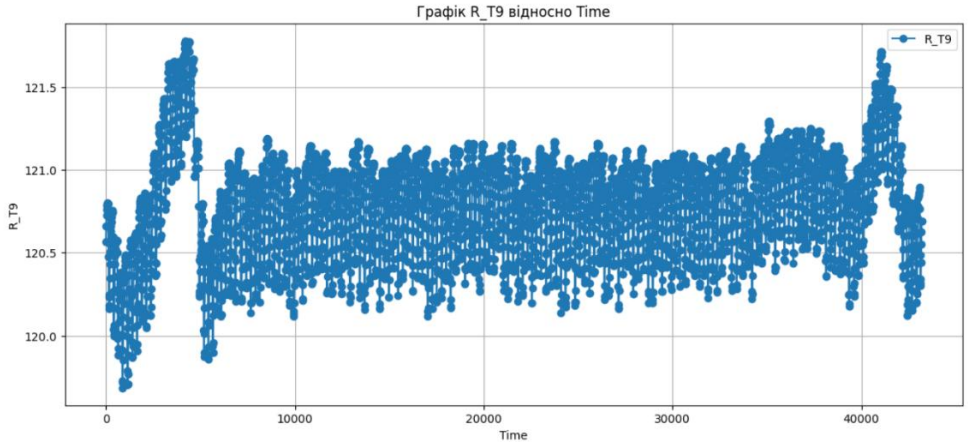

Рис. 2. Графік зміни температури над тринадцятою тарілкою колони концентрації

2. Перевірка концепції. Для перевірки працездатності моделі ВС на наявних даних та оцінки базових результатів було проведено попередню перевірку концепції (Proofof Concept) за допомогою пакета MATLAB. MATLAB — це пакет прикладних програм для числового аналізу, створений компанією Math Works. Вибір цього інструменту був зумовлений тим, що MATLAB володіє низкою необхідних для виконання цієї задачі функцій:

- є зручним інструментом для швидкої побудови й тестування моделей завдяки великій кількості вбудованих функцій, зокрема для роботи з нейронними мережами;
- дає змогу легко експериментувати з архітектурою моделі, кількістю затримок, кількістю нейронів у прихованому шарі та іншими параметрами моделі;
- має потужні інструменти для роботи з Time Series Split та іншими методами крос-валідації, що дає змогу перевірити стійкість моделі на різних наборах даних.

Реалізація NARX моделі в середовищі MATLAB із застосуванням Time Series Split для 5-foldcross-validation показала результати (рис. 3—5), на основі яких можна зробити такі висновки: усі чотири оброблені фолди демонструють низькі значення Train Performance (варіант критерію 1), що вказує на ефективне навчання моделі на тренувальних даних. Графіки Performance (рис. 4) відображають залежність значення MSE (по вертикалі) від кількості епох (по горизонталі) для навчальних і тестових вибірок в кожному фолді. Графіки регресії (рис. 5) показують реальні значення цільової змінної по вертикалі та передбачені по горизонталі. MATLA порівнює їх кореляцію та розраховує коефіцієнт кореляції дійсних і передбачених значень для оцінки

адекватності моделі: близькість точок до лінії $Y = T$ і високий коефіцієнт кореляції вказують на якість моделі. Результати Test Performance значно варіюються, значення похибки на першому фолді є значно вищими за значення в наступних фолдах, що може бути наслідком появи нових трендів на тестових даних, вплив яких зменшується зі збільшенням кількості навчальних даних моделі.

```
>> NARX_TimeSeriesSplit
Processing fold 1...
Fold 1 -> Train Performance: 0.000064, Test Performance: 0.075339
Processing fold 2...
Fold 2 -> Train Performance: 0.000063, Test Performance: 0.006470
Processing fold 3...
Fold 3 -> Train Performance: 0.000047, Test Performance: 0.006698
Processing fold 4...
Fold 4 -> Train Performance: 0.000081, Test Performance: 0.001071
Processing fold 5...
Not enough data points left for this fold. Skipping...
Average Train Performance: 0.000064
Average Test Performance: 0.022394
```

Рис. 3. Результати роботи NARX з Time Series Split для 5-fold cross-validation в MATLAB

Загалом, середнє значення Test Performance становить 0,022394, що вказує на задовільне узагальнення моделі, але з потенційними варіаціями залежно від частини даних. На четвертому фолді, коли модель отримала достатню варіативність даних для навчання, похибка на тестових даних становить всього 0,001, з коефіцієнтом кореляції 0,995, що вказує на те, що модель дуже добре відображає зв'язок між вхідними даними та цільовими значеннями.

Отримані результати вказують на перспективність подальшого моделювання.

3. Моделювання. Моделювання здійснювалося з використанням мови Python та його бібліотек в середовищі Google Colab. Було здійснено порівняння ефективності прогнозування таких моделей: XGBoost, Decision Tree, NARX, Gradient Boosting та RNN. Всі вони відповідають таким критеріям:

- можливість роботи з даними часових рядів;
- здатність працювати з нелінійними залежностями в даних;
- здатність виявляти складні залежності між вхідними змінними та цільовими показниками;
- можливість роботи з багатовимірними даними.

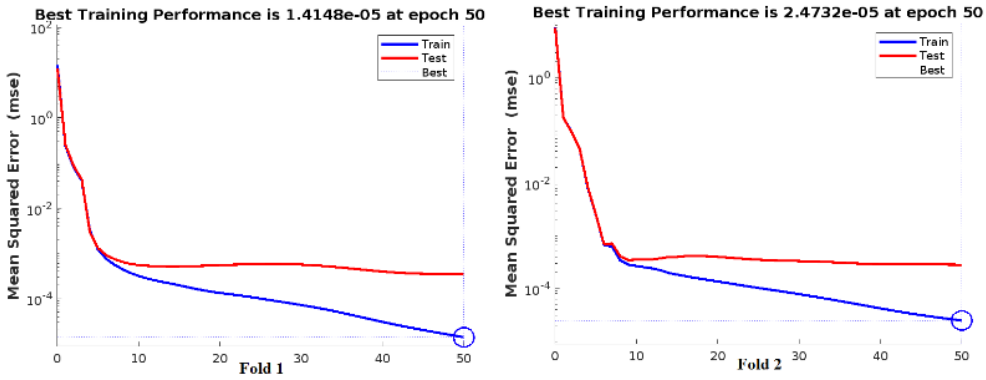

Рис. 4. Графіки Performance для NARX в MATLAB

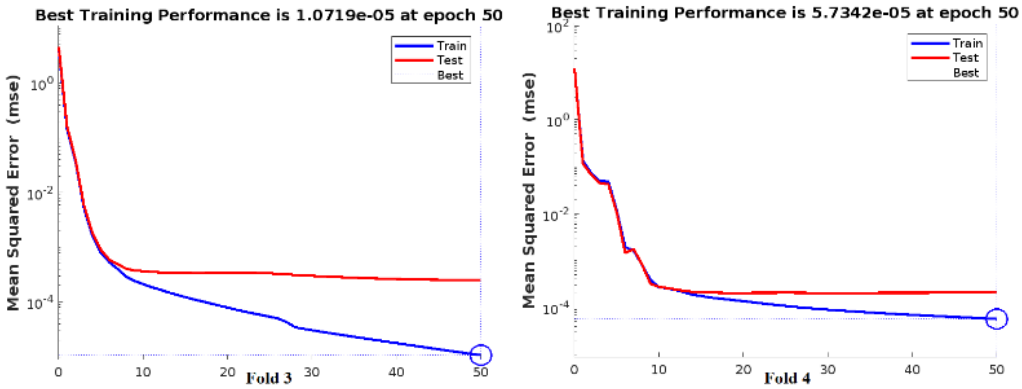

Рис. 4 (продовження). Графіки Performance для NARX в MATLAB

Відповідно до методу Time Series Split дані були розділені на п’ять навчальних і тестових вибірок зі збереженням часових послідовностей. З кожним fold кількість даних для навчання моделі збільшується, забезпечуючи донавчання моделі (табл. 1). Кожна з наведених моделей була навчена та оцінена відповідно до даних з Time Series Split.

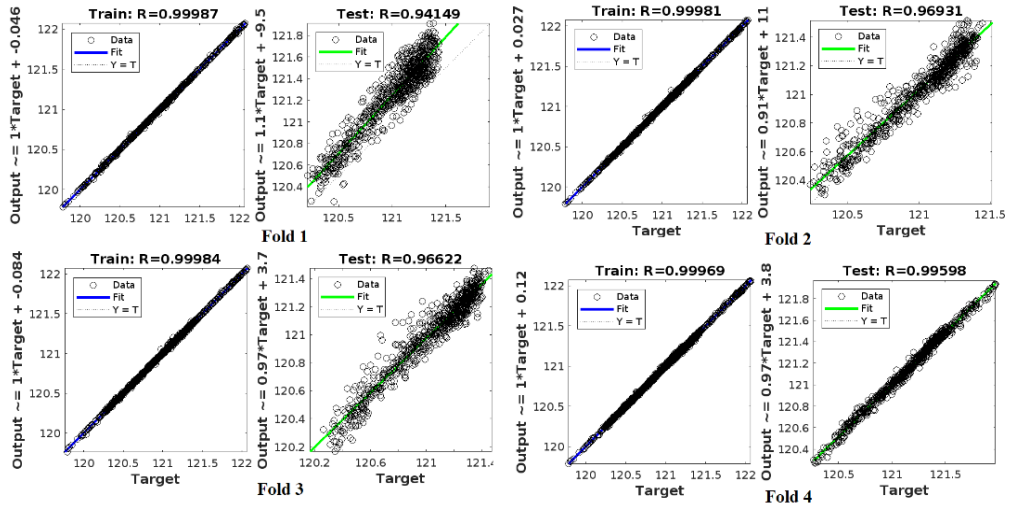

Рис. 5. Графіки регресії для NARX в MATLAB

Таблиця 1. **Розбиття даних на folds**

	Fold 1	**Fold 2**	**Fold 3**	**Fold 4**	**Fold 5**
Розмір навчального набору	603	1202	1801	2400	2999
Розмір валідаційного набору	599	599	599	599	599

XGBoost. XGBoost (eXtreme Gradient Boosting) — це бібліотека з відкритим вихідним кодом, яка використовується в машинному навчанні та надає функціональність для вирішення завдань, пов'язаних із регуляризацією градієнтного бустингу. Бібліотека показала такі результати на тестових даних (табл. 2).

Таблиця 2. **Результати передбачення XGBoost на тестових даних**

	MSE	**R^2**	**AIC**
Fold 1	0,00048707	0,9719	46431,37
Fold 2	0,00081122	0,9523	46086,68
Fold 3	0,00014358	0,9913	45699,66
Fold 4	0,00019493	0,9876	45882,83
Fold 5	0,00021514	0,9917	45941,92

Для відображення та порівняння взаємної динаміки всіх представлених метрик моделі було побудовано нормалізований графік за різними критеріями — це графічне представлення даних, де кожен критерій (або змінна) приведений до єдиної шкали, зазвичай у діапазоні від 0 до 1. Такий підхід дає змогу порівнювати величини, які мають різні одиниці вимірювання, масштаби або діапазони.

Нормалізований графік метрик показує їх відносні значення для кожного показника (рис. 6). Висока точність (низький MSE та високий R^2) вказує на те, що XGBoost змогла виявити складні патерни в даних. Проте високі значення AIC показують значну складність моделі.

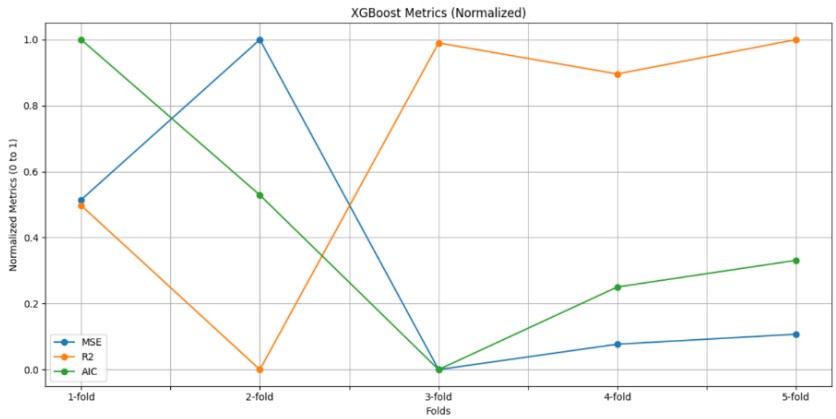

Рис. 6. Нормалізований графік метрик XGBoost

Порівняння фактичних і прогнозованих значень для тестових та валідаційних даних моделлю XGBoost (рис. 7) показує значний рівень відповідності На навчальних даних прогнозовані результати відображають майже ідеальне передбачення, що проявляється перекриттям графіка фактичних значень графіком передбачених.

Decision Tree. Decision Tree (Decision Tree Regressor) — це модель регресії на основі рішення дерев (decision trees), яка використовується для прогнозування безперервних цільових змінних. Вона належить до класу жадібних алгоритмів (greedy algorithm), які будують дерево, розділяючи дані на підмножини, що мінімізують помилку в кожному вузлі. Decision Tree Regressor реалізовано у бібліотеці scikit-learn.

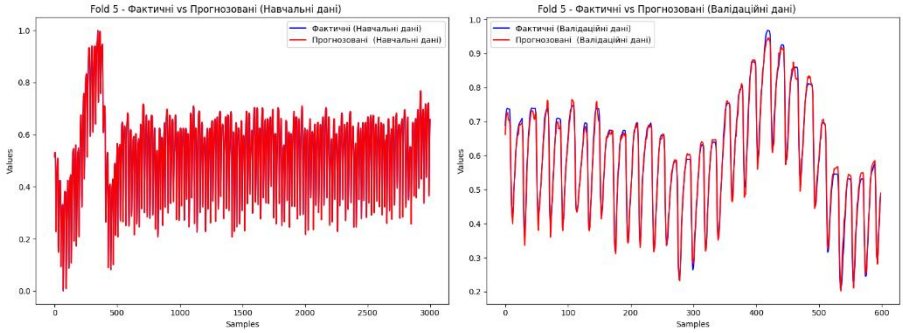

Рис. 7. Порівняння фактичних і прогнозованих значень для тестових та валідаційних даних моделлю XGBoost

Аналіз результатів роботи Decision Treeна тестових даних (табл. 3) вказує на її нижчу точність порівняно з XGBoost (рис. 8, 9), а від'ємні значення критерію AIC можуть свідчити про непридатність цього критерію для оцінки моделі такого типу, оскільки дерева рішень не є ймовірнісними моделями.

Таблиця 3. **Результати передбачення Decision Tree на тестових даних**

	MSE	**R^2**	**AIC**
Fold 1	0,00143358	0,9174	– 3852,00
Fold 2	0,00098765	0,9433	– 4075,19
Fold 3	0,00061588	0,9625	– 4358,08
Fold 4	0,00059168	0,9625	– 4382,09
Fold 5	0,00113648	0,9561	– 3991,11

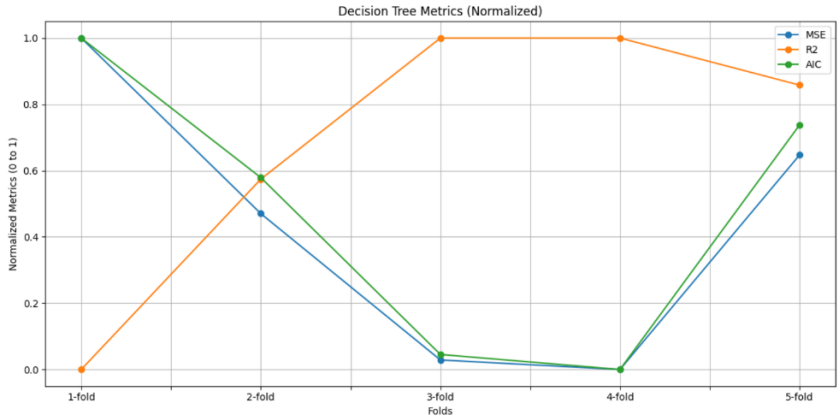

Рис. 8. Нормалізований графік метрик Decision Tree

NARX. NARX (Nonlinear Autoregressive with Exogenous Inputs) — це нелінійна модель авторегресії із зовнішніми входами, яка використовується для прогнозування часових рядів. Вона базується на припущенні, що поточне значення залежить від попередніх значень цільової змінної (авторегресія) та від попередніх значень зовнішніх вхідних змінних (екзогенні входи).

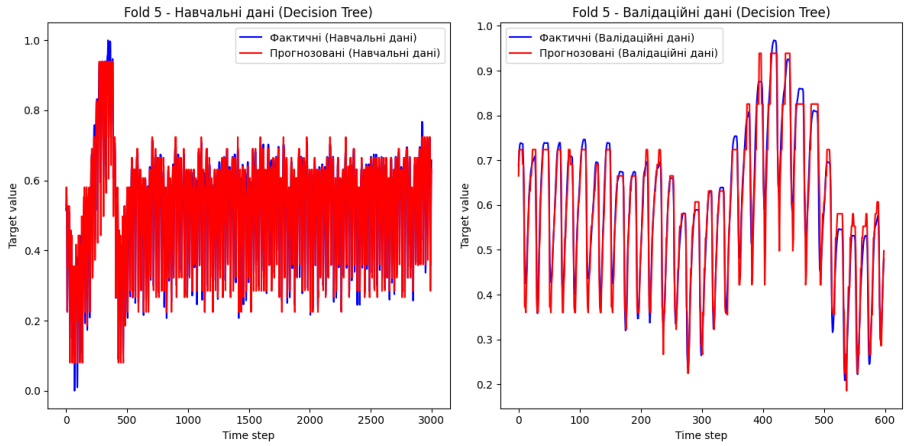

Рис. 9. Порівняння фактичних і прогнозованих значень для тестових та валідаційних даних моделлю Decision Tree

Низький MSE та високий R^2в усіх фолдах передбачення NARX моделі (табл. 4) свідчать про хорошу здатність моделі до передбачення, з незначною варіабельністю між фолдами. Дані у Fold 5 містять специфічні аномалії, до яких NARX не змогла адаптуватися через тренувальні залежності, що спричинило незначне зниження точності моделі на цьому проміжку.

Таблиця 4. ***Результати передбачення NARX на тестових даних***

	MSE	**R²**	**AIC**
Fold 1	0,00130916	0,9246	19577,62
Fold 2	0,00105109	0,9396	19446,10
Fold 3	0,00098448	0,9401	19406,88
Fold 4	0,00080229	0,9491	19284,30
Fold 5	0,00162684	0,9372	19707,75

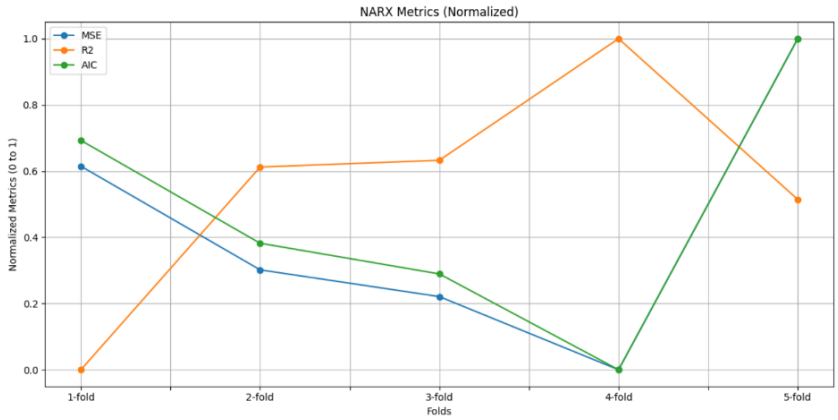

Рис. 10. Нормалізований графік метрик NARX

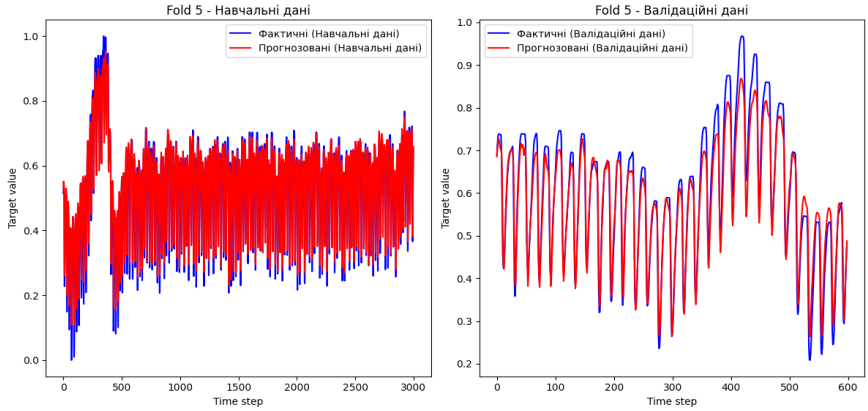

Рис. 11. Порівняння фактичних і прогнозованих значень для тестових та валідаційних даних моделлю NARX

RNN. RNN (Recurrent Neural Networks) — це тип нейронних мереж, які використовуються для обробки послідовних даних (наприклад, часових рядів, тексту, аудіо). Основна особливість RNN полягає в тому, що вони мають петлі зворотного зв'язку, які дають змогу моделі «запам'ятовувати» інформацію про попередні елементи послідовності. У Keras модуль SimpleRNN є базовим типом рекурентного шару. Кожен його нейрон має зв'язок із попереднім станом. SimpleRNN на кожному кроці часу враховує поточний вхід і попередній стан, оновлюючи інформацію про внутрішній стан, що дає змогу моделі враховувати часову залежність.

Модель показує значне покращення результатів на кожному наступному фолді. Це може свідчити про навчання на різних частинах даних і покращення здатності моделі до узагальнення. Незважаючи на різницю в результатах між фолдами, всі метрики показують досить хороші значення, що свідчить про стабільність моделі.

Таблиця 5. **Результати передбачення RNN на тестових даних**

	MSE	**R²**	**AIC**
Fold 1	0,00129290	0,9269	1324,78
Fold 2	0,00068793	0,9611	1046,05
Fold 3	0,00055452	0,9663	818,55
Fold 4	0,00041292	0,9739	642,23
Fold 5	0,00036811	0,9858	573,53

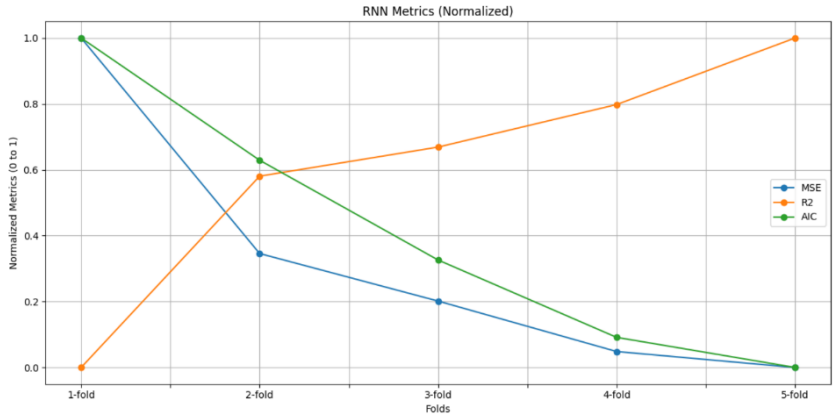

Рис. 12. Нормалізований графік метрик RNN

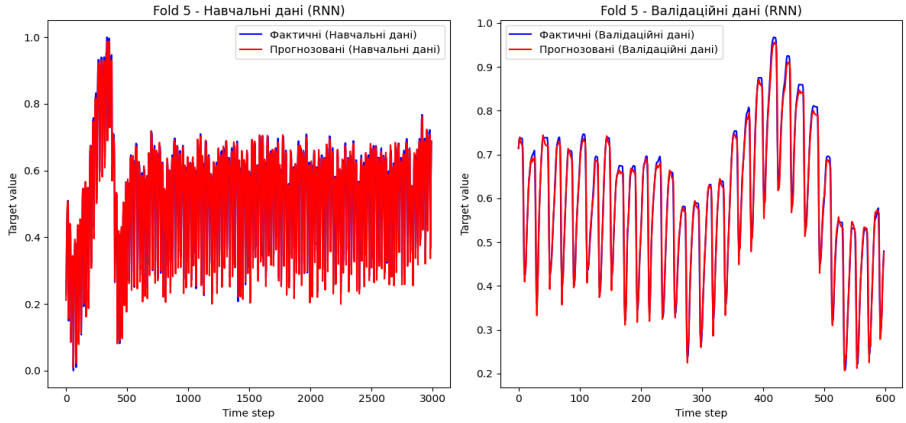

Рис. 13. Порівняння фактичних і прогнозованих значень для тестових та валідаційних даних моделлю RNN

Gradient Boosting. Gradient Boosting — це модель для регресії, що базується на техніці градієнтного бустингу. Вона використовується для побудови моделей регресії через послідовне тренування слабких моделей (зазвичай дерев рішень) на помилках попередніх моделей. На кожному кроці тренується нове дерево рішень, яке намагається зменшити помилку попередніх дерев. Це робиться шляхом оптимізації градієнта функції втрат.

Модель демонструє покращення результатів за MSE і R^2 в більшості фолдів, що свідчить про покращення якості передбачень на кожному наступному кроці. Як і в ситуації з NARX-моделлю, специфічні аномалії у Fold 5 дещо знижують точність прогнозування моделі.

Таблиця 6. **Результати передбачення Gradient Boosting на тестових даних**

	MSE	**R^2**	**AIC**
Fold 1	0,00048217	0,9722	3258,95
Fold 2	0,00024659	0,9858	3215,35
Fold 3	0,00021137	0,9871	3188,75
Fold 4	0,00015264	0,9903	3176,04
Fold 5	0,00027960	0,9892	3448,75

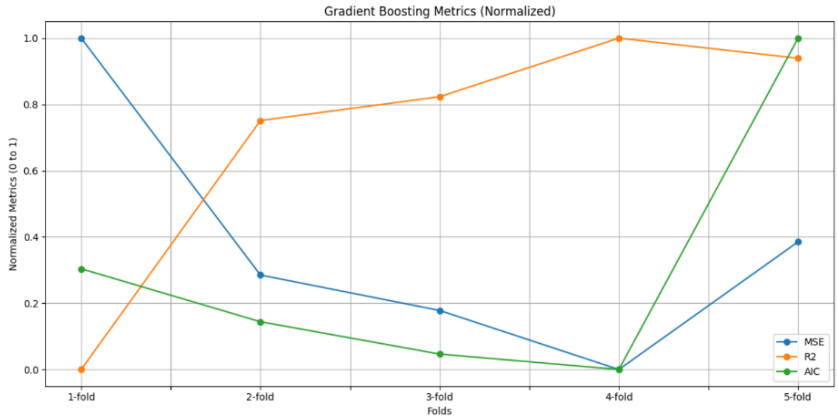

Рис. 14. Нормалізований графік метрик Gradient Boosting

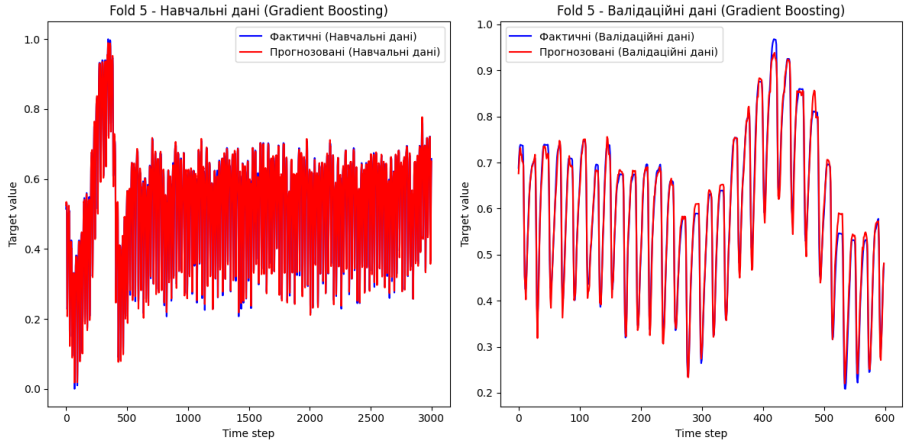

Рис. 15. Порівняння фактичних і прогнозованих значень для тестових та валідаційних даних моделлю Gradient Boosting

4. Порівняння результатів. Усереднені результати всіх моделей за всіма фолдами представлені в табл. 7. Незважаючи на те, що всі представлені моделі показали позитивні результати передбачення, деякі з них продемонстрували значну різницю в точності. Так, моделі XGBoost, Gradient Boosting та RNN мають на порядок нижчу середньоквадратичну похибку (рис.16), та вищу здатність моделей пояснювати варіацію даних (рис.17), ніж Decision Tree та NARX.

Таблиця 7. **Усереднені порівняльні результати моделей за всіма фолдами**

	MSE	R^2	AIC
Gradient Boosting	0,00027447	0,9849	3257,57
XGBoost	0,00026294	0,9853	46008,49
RNN	0,00068793	0,9611	881,03
DecisionTree	0,00095305	0,9484	– 4131,70
NARX	0,00115477	0,9381	19484,53

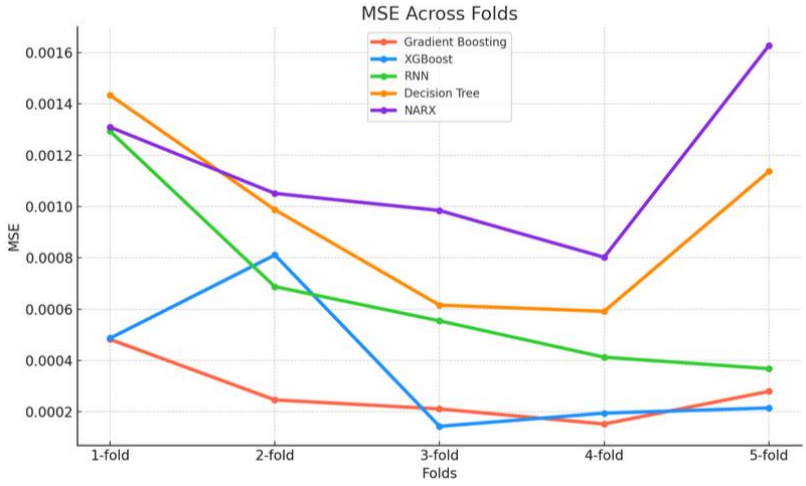

Рис. 16. Порівняння значень MSE для всіх моделей

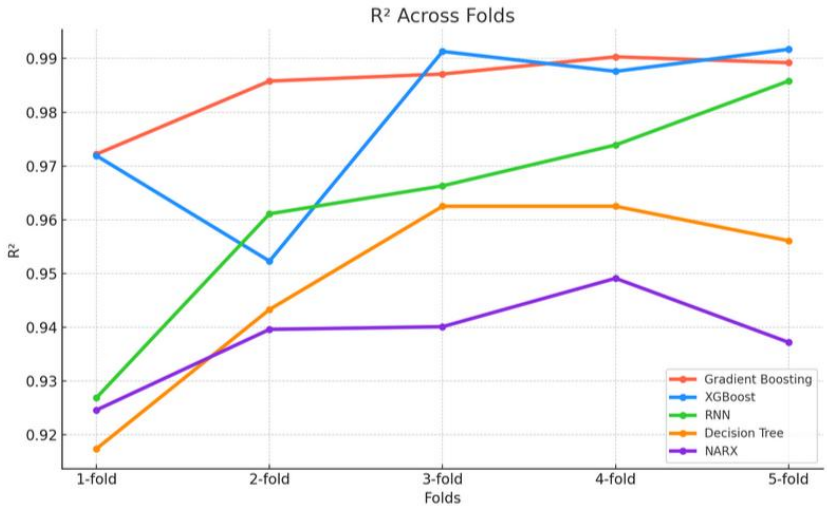

Рис. 17. Порівняння значень R^2 для всіх моделей

Порівняння балансу між точністю і складністю в моделях за допомогою інформаційного критерію Акаіке показує підвищену складність моделей XGBoostта NARX. Враховуючи те, що Decision Tree не відповідає вимогам використання AIC, пов'язаним з імовірнісною природою моделі, що спотворює результати його аналізу наявністю від'ємних значень. Найкращими за цим показником моделями є Gradient Boosting та RNN (рис. 18).

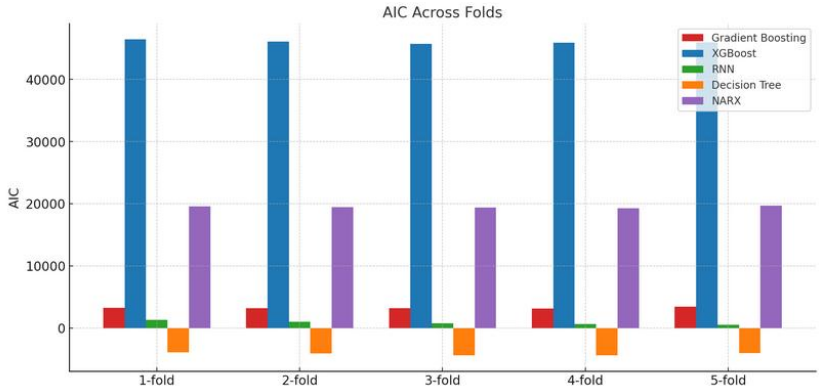

Рис. 18. Порівняння значень AIC для всіх моделей

Висновки. У ході дослідження було розроблено моделі віртуальних сенсорів на базі методів машинного навчання та проведено попередню перевірку працездатності розробленої концепції за допомогою засобів MATLAB. Використовуючи бібліотеки Python, було реалізовано навчання і тестування моделей XGBoost, Decision Tree, NARX, Gradient Boosting та RNN. Для оцінки їх ефективності здійснено порівняльний аналіз за допомогою метрик якості моделей: середньоквадратичної похибки, коефіцієнта детермінації та інформаційного критерію Акаіке.

Відповідно до результатів дослідження найкращою моделлю для аналізу перебігу процесів дистиляційного відділення спиртового виробництва в складі ВС є Gradient

Boosting. Вона показала стабільну роботу на всіх тестових наборах даних, використаних для валідації. При цьому її максимальна середньоквадратична похибка склала 0,00048, при якій модель здатна пояснити 97,2% варіації цільової змінної. Водночас Gradient Boosting показує оптимальні значення складності порівняно з іншими моделями згідно з критерієм Акаіке.

Обрану модель ВС планується інтегрувати в систему моніторингу дистиляційного відділення як компонент цифрового двійника, що дасть змогу підвищити ефективність управління технологічним процесом і забезпечити надійний інтелектуальний аналіз даних у реальному часі.

ЛІТЕРАТУРА

1. Kano, M., & Fujiwara, K. (2013). Virtual sensing technology in process industries: trends and challenges revealed by recent industrial applications. *Journal of chemical engineering of Japan*, 46(1), 1—17.

2. Martin, D., Kühl, N., & Satzger, G. (2021). Virtual sensors. *Business & Information Systems Engineering*, 63, 315—323.

3. Cristaldi, L., Ferrero, A., Macchi, M., Mehrafshan, A., & Arpaia, P. (2020, June). Virtual sensors: a tool to improve reliability. In 2020 IEEE International Workshop on Metrology for Industry 4.0 & IoT (pp. 142—145). IEEE.

4. Goodwin, G. C. (1999, February). Evaluating the performance of virtual sensors. In 1999 Information, Decision and Control. Data and Information Fusion Symposium, Signal Processing and Communications Symposium and Decision and Control Symposium. Proceedings (Cat. No. 99EX251) (pp. 5—12). IEEE.

5. Li, H., Yu, D., & Braun, J. E. (2011). A review of virtual sensing technology and application in building systems. *Hvac&R Research*, 17(5), 619—645.

6. Zhang, M. Z., Wang, L. M., & Xiong, S. M. (2020). Using machine learning methods to provision virtual sensors in sensor-cloud. *Sensors*, 20(7), 1836.

7. Hirsenkorn, N., Hanke, T., Rauch, A., Dehlink, B., Rasshofer, R., & Biebl, E. (2016). Virtual sensor models for real-time applications. *Advances in Radio Science*, 14, 31—37.

8. Kullaa, J. (2017). Development of virtual sensors to increase the sensitivity to damage. *Procedia engineering*, 199, 1937—1942.

9. Liu, L., Kuo, S. M., & Zhou, M. (2009, March). Virtual sensing techniques and their applications. In 2009 International Conference on Networking, Sensing and Control (pp. 31—36). IEEE.

10. Dimitrov, N., & Göçmen, T. (2022). Virtual sensors for wind turbines with machine learning-based time series models. *Wind Energy*, 25(9), 1626—1645.

11. Stavropoulos, G., Violos, J., Tsanakas, S., & Leivadeas, A. (2023). Enabling artificial intelligent virtual sensors in an IoT environment. *Sensors*, 23(3), 1328.

12. Wei, C., Chen, J., Song, Z., & Chen, C. I. (2018). Development of self-learning kernel regression models for virtual sensors on nonlinear processes. IEEE *Transactions on Automation Science and Engineering*, 16(1), 286—297.

13. Kim, W., & Braun, J. E. (2020). Development, implementation, and evaluation of a fault detection and diagnostics system based on integrated virtual sensors and fault impact models. *Energy and Buildings*, 228, 110368.